```
FFFFFFFFFFFFFF      111            111          AAAAAAAAA
FFFFFFFFFFFFFF      111            111          AAAAAAAAA
FFFFFFFFFFFFFF      111            111          AAAAAAAAA
FFF                111111         111111       AAA      AAA
FFF                111111         111111       AAA      AAA
FFF                111111         111111       AAA      AAA
FFF                 111            111         AAA      AAA
FFF                 111            111         AAA      AAA
FFFFFFFFFFF         111            111         AAA      AAA
FFFFFFFFFFF         111            111         AAA      AAA
FFFFFFFFFFF         111            111         AAAAAAAAAAAAAAAA
FFF                 111            111         AAAAAAAAAAAAAAAA
FFF                 111            111         AAAAAAAAAAAAAAAA
FFF                 111            111         AAA      AAA
FFF                 111            111         AAA      AAA
FFF                 111            111         AAA      AAA
FFF              11111111        11111111      AAA      AAA
FFF              11111111        11111111      AAA      AAA
FFF              11111111        11111111      AAA      AAA
```

```
GGGGGGG   EEEEEEEEEE   TTTTTTTTTT   FFFFFFFFFF   IIIIII   BBBBBBBB
GGGGGGGG  EEEEEEEEEE   TTTTTTTTTT   FFFFFFFFFF   IIIIII   BBBBBBBB
GG        EE               TT       FF              II    BB      BB
GG        EE               TT       FF              II    BB      BB
GG        EE               TT       FF              II    BB      BB
GG        EEEEEEE          TT       FFFFFFFF        II    BBBBBBBB
GG        EEEEEEE          TT       FFFFFFFF        II    BBBBBBBB
GG  GGGGGG EE              TT       FF              II    BB      BB
GG  GGGGGG EE              TT       FF              II    BB      BB
GG      GG EE              TT       FF              II    BB      BB
GG      GG EE              TT       FF              II    BB      BB     ....
GGGGGG    EEEEEEEEEE       TT       FF           IIIIII   BBBBBBBB       ....
GGGGGG    EEEEEEEEEE       TT       FF           IIIIII   BBBBBBBB       ....


LL        IIIIII     SSSSSSSS
LL        IIIIII     SSSSSSSS
LL          II     SS
LL          II     SS
LL          II     SS
LL          II       SSSSSS
LL          II       SSSSSS
LL          II              SS
LL          II              SS
LL          II              SS
LL          II              SS
LLLLLLLLL   IIIIII   SSSSSSSS
LLLLLLLLL   IIIIII   SSSSSSSS
```

```
0001  0  MODULE GETFIB (
0002  0                 LANGUAGE (BLISS32),
0003  0                 IDENT = 'V04-000'
0004  0                 ) =
0005  1  BEGIN
0006  1
0007  1
0008  1  !*******************************************************************
0009  1  !*                                                                 *
0010  1  !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                        *
0011  1  !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.         *
0012  1  !*  ALL RIGHTS RESERVED.                                           *
0013  1  !*                                                                 *
0014  1  !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0015  1  !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE *
0016  1  !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER *
0017  1  !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0018  1  !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY *
0019  1  !*  TRANSFERRED.                                                   *
0020  1  !*                                                                 *
0021  1  !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE *
0022  1  !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT *
0023  1  !*  CORPORATION.                                                   *
0024  1  !*                                                                 *
0025  1  !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS *
0026  1  !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.        *
0027  1  !*                                                                 *
0028  1  !*                                                            .    *
0029  1  !*******************************************************************
0030  1
0031  1  !++
0032  1  !
0033  1  ! FACILITY:  F11ACP Structure Level 1
0034  1  !
0035  1  ! ABSTRACT:
0036  1  !
0037  1  !     This routine obtains the address of the FIB for this operation.
0038  1  !
0039  1  ! ENVIRONMENT:
0040  1  !
0041  1  !     STARLET operating system, including privileged system services
0042  1  !     and internal exec routines.
0043  1  !
0044  1  !--
0045  1  !
0046  1  !
0047  1  ! AUTHOR:  Andrew C. Goldstein,  CREATION DATE:  7-Jan-1977  01:02
0048  1  !
0049  1  ! MODIFIED BY:
0050  1  !
0051  1  !     V03-001 LMP0219        L. Mark Pilant,        24-Mar-1984  23:15
0052  1  !             Preset FIB$L_ACL_STATUS to SS$_NORMAL.
0053  1  !
0054  1  !     A0100   ACG0001        Andrew C. Goldstein,  10-Oct-1978  20:01
0055  1  !             Previous revision history moved to F11A.REV
0056  1  !
0057  1  !**
```

GETFIB
V04-000

N 11
16-Sep-1984 01:05:44   VAX-11 Bliss-32 V4.0-742   Page 2
14-Sep-1984 12:29:37   DISK$VMSMASTER:[F11A.SRC]GETFIB.B32;1   (1)

```
:   58           0058  1
:   59           0059  1
:   60           0060  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
:   61           0061  1 REQUIRE 'SRC$:FCPDEF.B32';
```

```
000
```

GETFIB
V04-000

B 12
16-Sep-1984 01:05:44   VAX-11 Bliss-32 V4.0-742   Page 3
14-Sep-1984 12:29:37   DISK$VMSMASTER:[F11A.SRC]GETFIB.B32;1   (2)

```
 63    0376  1  GLOBAL ROUTINE GET_FIB (ABD) =
 64    0377  1
 65    0378  1  !++
 66    0379  1  !
 67    0380  1  !  FUNCTIONAL DESCRIPTION:
 68    0381  1  !
 69    0382  1  !      This routine obtains the address of the FIB for this operation.
 70    0383  1  !      It copies the FIB from the buffer packet into local storage
 71    0384  1  !      and zero extends it to maximum length.
 72    0385  1  !
 73    0386  1  !  CALLING SEQUENCE:
 74    0387  1  !      GET_FIB (ARG1)
 75    0388  1  !
 76    0389  1  !  INPUT PARAMETERS:
 77    0390  1  !      ARG1: buffer descriptor list
 78    0391  1  !
 79    0392  1  !  IMPLICIT INPUTS:
 80    0393  1  !      CURRENT_WINDOW: address of user's window or 0
 81    0394  1  !      IO_PACKET: address of user's I/O packet
 82    0395  1  !
 83    0396  1  !  OUTPUT PARAMETERS:
 84    0397  1  !      NONE
 85    0398  1  !
 86    0399  1  !  IMPLICIT OUTPUTS:
 87    0400  1  !      NONE
 88    0401  1  !
 89    0402  1  !  ROUTINE VALUE:
 90    0403  1  !      address of FIB
 91    0404  1  !
 92    0405  1  !  SIDE EFFECTS:
 93    0406  1  !      file ID may be written into FIB
 94    0407  1  !      channel window pointer write-back inhibited
 95    0408  1  !      result string buffers zeroed
 96    0409  1  !
 97    0410  1  !--
 98    0411  1
 99    0412  2  BEGIN
100    0413  2
101    0414  2  MAP
102    0415  2      ABD                : REF BBLOCKVECTOR [,ABD$C_LENGTH];
103    0416  2                                     ! buffer descriptors
104    0417  2
105    0418  2  LOCAL
106    0419  2      FCB                : REF BBLOCK,    ! FCB of file
107    0420  2      FIBL;                             ! length of user FIB
108    0421  2
109    0422  2  EXTERNAL
110    0423  2      LOCAL_FIB          : BBLOCK,       ! internal copy of user FIB
111    0424  2      IO_PACKET          : REF BBLOCK,   ! I/O packet of this operation
112    0425  2      PRIMARY_FCB        : REF BBLOCK,   ! FCB of current file
113    0426  2      CURRENT_FIB        : REF BBLOCK,   ! pointer to current FIB in use
114    0427  2      CURRENT_WINDOW     : REF BBLOCK;   ! user's window
115    0428  2
116    0429  2
117    0430  2  ! Get the length of the user-supplied FIB. If there is a window,
118    0431  2  ! and there is no user FIB, use the file ID from
119    0432  2  ! the window's FCB. Also use the FCB's file ID if the file number
```

GETFIB
V04-000

C 12
16-Sep-1984 01:05:44   VAX-11 Bliss-32 V4.0-742   Page 4
14-Sep-1984 12:29:37   DISK$VMSMASTER:[F11A.SRC]GETFIB.B32;1   (2)

```
 120    0433    2 ! in the user FIB is zero.
 121    0434    2 !
 122    0435    2
 123    0436    2 FIBL = .ABD[ABD$C_FIB, ABD$W_COUNT];
 124    0437    2
 125    0438    2 CH$COPY (.FIBL
 126    0439    2          .ABD[ABD$C_FIB, ABD$W_TEXT] + ABD[ABD$C_FIB, ABD$W_TEXT] + 1,
 127    0440    2          0
 128    0441    2          FIB$C_LENGTH,
 129    0442    2          LOCAL_FIB);
 130    0443    2 CURRENT_FIB = LOCAL_FIB;
 131    0444    2
 132    0445    2 LOCAL_FIB[FIB$L_ACL_STATUS] = SS$_NORMAL;                 ! Preset to success
 133    0446    2
 134    0447    2 IF .CURRENT_WINDOW NEQ 0
 135    0448    2 THEN
 136    0449    2     BEGIN
 137    0450    2     FCB = .CURRENT_WINDOW[WCB$L_FCB];
 138    0451    3     IF .LOCAL_FIB[FIB$W_FID_NUM] EQL 0
 139    0452    3     THEN CH$MOVE (FIB$S_FID, FCB[FCB$W_FID], LOCAL_FIB[FIB$W_FID]);
 140    0453    3
 141    0454    3 ! If the file ID in the FIB does not match that in the FCB, this operation
 142    0455    3 ! is not on the open file; clear the FCB and window addresses (except in
 143    0456    3 ! the case of a DEACCESS, in which we force the file ID to that of the open
 144    0457    3 ! file and signal an error).
 145    0458    3 !
 146    0459    3
 147    0460    3     IF .LOCAL_FIB[FIB$W_FID_NUM] NEQ .FCB[FCB$W_FID_NUM]
 148    0461    3     OR .LOCAL_FIB[FIB$W_FID_RVN] NEQ .FCB[FCB$W_FID_RVN]
 149    0462    3     THEN
 150    0463    4         BEGIN
 151    0464    4         IF .IO_PACKET[IRP$V_FCODE] EQL IO$_DEACCESS
 152    0465    4         THEN
 153    0466    5             BEGIN
 154    0467    5             CH$MOVE (FIB$S_FID, FCB[FCB$W_FID], LOCAL_FIB[FIB$W_FID]);
 155    0468    5             ERR_STATUS (SS$_BADPARAM);
 156    0469    5             END
 157    0470    4         ELSE
 158    0471    5             BEGIN
 159    0472    5             CURRENT_WINDOW = 0;
 160    0473    5             PRIMARY_FCB = 0;
 161    0474    5             END;
 162    0475    3         END;
 163    0476    3     END
 164    0477    3
 165    0478    3 ! If there is no file open, there must be a minimum FIB.
 166    0479    3 !
 167    0480    3
 168    0481    2 ELSE
 169    0482    3     BEGIN
 170    0483    3     IF .FIBL LSS FIB$C_ACCDATA
 171    0484    3     THEN ERR_EXIT (SS$_INSFARG);
 172    0485    3     END;
 173    0486    2
 174    0487    2 ! If the directory ID is -1,-1, convert it to 4,4,0 to be compatible with
 175    0488    2 ! the old RSX MFD kluge.
 176    0489    2 !
```

```
 ; 177    0490 2
 ; 178    0491 2      IF .LOCAL_FIB[FIB$W_DID_NUM] EQL 65535
 ; 179    0492 2      AND .LOCAL_FIB[FIB$Q_DID_SEQ] EQL 65535
 ; 180    0493 2      THEN
 ; 181    0494 2           BEGIN
 ; 182    0495 3           LOCAL_FIB[FIB$W_DID_NUM] = 4;
 ; 183    0496 3           LOCAL_FIB[FIB$W_DID_SEQ] = 4;
 ; 184    0497 3           LOCAL_FIB[FIB$W_DID_RVN] = 0;
 ; 185    0498 2           END;
 ; 186    0499 2
 ; 187    0500 2      RETURN LOCAL_FIB;
 ; 188    0501 2
 ; 189    0502 1 END;                                   ! end of routine GET_FIB


                                        .TITLE   GETFIB
                                        .IDENT   \V04-000\

                                        .EXTRN   LOCAL_FIB, IO_PACKET
                                        .EXTRN   PRIMARY_FCB, CURRENT_FIB
                                        .EXTRN   CURRENT_WINDOW, USER_STATUS

                                        .PSECT   $CODE$,NOWRT,2

                       01FC 00000       .ENTRY   GET_FIB, Save R2,R3,R4,R5,R6,R7,R8    ; 0376
         58 00000000G  00   9E 00002    MOVAB    USER_STATUS, R8
         57       0000G CF   9E 00009    MOVAB    LOCAL_FIB+4, R7
         50        04  AC   D0 0000E    MOVL     ABD, R0                              ; 0436
         56        0A  A0   3C 00012    MOVZWL   10(R0), FIBL
         51        08  A0   9E 00016    MOVAB    8(R0), R1                            ; 0439
         50            61   3C 0001A    MOVZWL   (R1), R0
0040 8F          00   01 A140 56 2C 0001D   MOVC5  FIBL, 1(R1)[R0], #0, #64, LOCAL_FIB  ; 0438
                       A7      00026
              0000G CF  FC  A7  9E 00028    MOVAB    LOCAL_FIB, CURRENT_FIB           ; 0443
                   30  A7  01  D0 0002E    MOVL     #1, LOCAL_FIB+52                  ; 0445
                   50      0000G CF D0 00032  MOVL  CURRENT_WINDOW, R0               ; 0447
                       3E  13 00037    BEQL     4$
              56   18  A0  D0 00039    MOVL     24(R0), FCB                          ; 0450
                   67  B5 0003D    TSTW     LOCAL_FIB+4                              ; 0451
                   05  12 0003F    BNEQ     1$
         67    24  A6   06  28 00041    MOVC3    #6, 36(FCB), LOCAL_FIB+4            ; 0452
         24    A6   67   B1 00046 1$:  CMPW     LOCAL_FIB+4, 36(FCB)                ; 0460
                   07  12 0004A    BNEQ     2$
         28  A6   04  A7  B1 0004C    CMPW     LOCAL_FIB+8, 40(FCB)                ; 0461
                   2E  13 00051    BEQL     5$
         50    0000G CF  D0 00053 2$: MOVL     IO_PACKET, R0                       ; 0464
34   20  A0    06   00  ED 00058    CMPZV    #0, #6, 32(R0), #52
                   0D  12 0005E    BNEQ     3$
         67    24  A6   06  28 00060    MOVC3    #6, 36(FCB), LOCAL_FIB+4           ; 0467
              19   68  E9 00065    BLBC     USER_STATUS, 5$                        ; 0468
              68   14  B0 00068    MOVW     #20, USER_STATUS
              14   11 0006B    BRB      5$
         0000G CF  D4 0006D 3$: CLRL     CURRENT_WINDOW                            ; 0464
         0000G CF  D4 00071    CLRL     PRIMARY_FCB                                ; 0472
              0A   11 00075    BRB      5$                                          ; 0473
         0A     56  D1 00077 4$: CMPL     FIBL, #10                                ; 0460
              05   18 0007A    BGEQ     5$                                          ; 0483
```

```
                                    0114   8F  BF 0007C         CHMU    #276                                        ; 0484
                                           04 00080             RET                                                 ; 
                      FFFF   8F       06   A7  B1 00081  5$:     CMPW    LOCAL_FIB+10, #65535                        ; 0491
                                           13  12 00087         BNEQ    6$
                      FFFF   8F       08   A7  B1 00089          CMPW    LOCAL_FIB+12, #65535                        ; 0492
                                           0B  12 0008F         BNEQ    6$
                      06   A7 00040004  8F  D0 00091            MOVL    #262148, LOCAL_FIB+10                       ; 0495
                                      0A   A7  B4 00099         CLRW    LOCAL_FIB+14                                ; 0497
                                      50   FC  A7  9E 0009C 6$:  MOVAB   LOCAL_FIB, R0                              ; 0500
                                           04 000A0             RET                                                 ; 0502
```

; Routine Size: 161 bytes,   Routine Base: $CODE$ + 0000

```
; 190           0503  1
; 191           0504  1 END
; 192           0505  0 ELUDOM
```

```
;                          PSECT SUMMARY
;
;       Name                     Bytes                      Attributes
;
;   $CODE$                        161  NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)
```

```
;                    Library Statistics
;
;                                   -------- Symbols --------      Pages        Processing
;       File                        Total   Loaded   Percent      Mapped       Time
;
;   _$255$DUA28:[SYSLIB]LIB.L32;1   18619     24        0          1000         00:01.9
```

```
;                         COMMAND QUALIFIERS
;
;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:GETFIB/OBJ=OBJ$:GETFIB MSRC$:GETFIB/UPDATE=(ENH$:GETFIB)
```

```
; Size:          161 code + 0 data bytes
; Run Time:          00:07.4
; Elapsed Time:      00:27.9
; Lines/CPU Min:      4105
; Lexemes/CPU-Min: 15105
; Memory Used:  97 pages
; Compilation Complete
```

EXTFCB
LIS

DELFIL
LIS

DIRGET
LIS

EXTIDX
LIS

IODONE
LIS

LOCKDN
LIS

ENTER
LIS

GETREQ
LIS

GETTIM
LIS

DISPAT
LIS

INIFCP
LIS

DIRFCB
LIS

EXTHDR
LIS

DIRSCN
LIS

LOGDEL
LIS

LOCKDB
LIS

FIND
LIS

GETFIB
LIS

DIRACC
LIS

INIFCB
LIS

EXTDIR
LIS

EXTEND
LIS